Challenges and approaches for the visualization of movement trajectories in 3D geovirtual environments

Stefan Buschmann, Matthias Trapp, and Jürgen Döllner

Abstract— Visualization of trajectories and their attributes represents an essential functionality for spatio-temporal data visualization and analysis. Many visualization methods, however, focus mainly on sparse 2D movements or consider only the 2D components of movements. This paper is concerned with true 3D movement data, i.e., movements that take place in the three-dimensional space and which characteristics significantly depend an all dimensions. In this case, spatio-temporal visualization approaches need to map all three spatial dimensions together with required mappings for associated attributes. In this paper, visualization approaches for 3D movement data are described and their applications within 3D geovirtual environments, such as virtual 3D city or landscape models, are evaluated. Also, challenges for true 3D movement visualization methods are identified and approaches for the interactive visualization of 3D movement data within 3D geovirtual environments are proposed.

Index Terms—spatio-temporal data, trajectories, interactive 3D visualization, visual analytics

1 Introduction

In the discipline of spatio-temporal data analysis and visualization, a lot of research has been conducted on the visualization of temporal aspects of data. Research goals generally include the development of visualization methods that improve the human perception and understanding of temporal aspects in data sets, i.e., concurrency and temporal order of events [1]. For that, both the spatial and temporal aspects have to be visualized in a way that is as intuitive as possible for a user to understand. In addition to static mappings, animations and interactive exploration [2] can provide additional means for the understanding of temporal aspects.

In this paper we focus on moving-object data and their visualization within 3D geovirtual environments. A movement of an object can be described by its trajectory, which is the way it has traveled. Trajectories are usually represented by a number of points sampled at discrete time intervals and can be modelled by continuous line or curve segments that connect the sample points. In general, two approaches for the visualization of movement data can be distinguished: (1) direct visualization of trajectories and (2) visualization of aggregated data. While the first approach aims at visualizing trajectories directly, e.g., by polylines or curves, the second approach examines a set of related trajectories, e.g., by computing each trajectory's contribution to its neighborhood, and aggregates the results. Using this approach, the movement of objects in space can be aggregated spatially and interpreted as a *density map* [3], which can be used to determine local clusters and hot spots.

In addition to the spatial and temporal components of a movement, represented by its trajectory, movement data is often attributed. Those attributes can contain additional information and meta-data about the movement, e.g., the current direction and speed, or domain specific data of the specific use case, e.g., the type of vehicles. Therefore, visualization approaches also have to map these time-dependent attributes along with spatial and temporal information of a movement.

In many cases, the 2D geographic coordinates of movement data are sufficient for analysis and visualization. Depending on the use case, either only the 2D component of a movement has been tracked, or, while the data itself has a third dimension, the 3D component can be omitted. Considering for example the movement of pedestrians, cars, or ships, movement analysis and visualization are often performed on a 2D basis. When analyzing long range airplane movements, for example to determine flight traffic volumes on a geographical scale, the 3D component of the movement is frequently omitted.

In a growing number of applications, however, movements need to be tracked in the three-dimensional space since their movement characteristics significantly depend on all dimensions. For example, we focus on visualizing air craft trajectories around and between airports. Especially the detection of flight routes, common flight corridors and spatial and temporal hotspots demand for a true 3D analysis. Therefore, the actual 3D positions of aircrafts have to be maintained and expressed in a visualization. On the other hand, temporal aspects have to be conveyed as well. This poses a number of challenges for 3D visualization techniques of spatio-temporal data:

Visualization of temporal aspects: Since all three axes are used to display the spatial (3D) components of the data itself, there is no axis left that could be used to map the temporal information. Therefore, temporal information has to be mapped to other visual dimensions, such as color, width, or style, or has to be visualized in other forms, e.g., by means of animation.

Perspective distortion: When using perspective projections of 3D scenes onto planar surfaces, spatial positions appear perspectively distorted on the image. This effect is inevitable, but must be recognized since it can disturb the cognition of the displayed data. For example, trajectories may seem to overlap although in reality they do not. Also the perception of height and size in perspective images is difficult and prone to misinterpretation.

Occlusion and clutter: Another major problem posed by the rendering of virtual 3D scenes is occlusion [4]. Trajectories overlap each other, thereby occluding objects behind them. A large number of objects displayed at the same time also lead to visual clutter. These problems can be approached by filtering (hiding objects that are currently not of importance), selection (highlighting important objects), and transparency (making objects semi-transparent to partially visualize occluded objects).

In this paper, we describe several visualization approaches for movement data and explore their application within 3D geovirtual environments. We identify potentials and limitations resulting of the extention of those methods for true 3D movement data and discuss strategies to approach some of these limitations.

- *S. Buschmann: Hasso-Plattner-Institut Potsdam, e-mail: stefan.buschmann@hpi.uni-potsdam.de*
- *M. Trapp: Hasso-Plattner-Institut Potsdam, e-mail: matthias.trappt@hpi.uni-potsdam.de*
- *J. Döllner: Hasso-Plattner-Institut Potsdam, e-mail: juergen.doellner@hpi.uni-potsdam.de*

Manuscript received 31 March 2012; accepted 1 August 2012; posted online 14 October 2012; mailed on 5 October 2012.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

2 PRELIMINARIES

This section gives an overview of the domain and use case from which this research is derived. Also, the used data set and its data acquisition is desribed.

2.1 Use case and context

This research is focused on visualization and analysis of air traffic data, i.e., movements of air planes. Its aim is to develop a visual analytics (VA) framework for interactive investigation and exploration of movement data, that allows expert users to examine the movements of air planes and search for spatial and temporal phenomena. This is developed from a management and planning point of view and therefore targets at Air Traffic Management (ATM), which is concerned with analyzing past events and deriving insight from it as to understanding the cause of events and learning how to prevent them or optimizing processes in the future. It is not meant as an operational tool for Air Traffic Control (ATC), which has to work with real-time data under extreme safety measures.

Exemplary questions and interests for the analysis or air plane movements include:

Common flight paths: By investigating the distribution of movements over time and space, areas with higher and lower traffic can be identified. From this, common flight routes which are used often can be derived. Trajectories can be categorized according to those common flight routes.

Classification: A comparison of flight routes allows to identify similarities and differences between different type of flights. By correlating them to other flight attributes (e.g., departing and landing flights, or different air craft types), a classification of flights can be acchieved.

Discover typical and untypical patterns: Categorisation and clustering methods are employed to categorize trajectories. A further goal is to identify untypical movement behaviors and if possible correlate them to spatial or temporal circumstances.

Adherence to safety regulations: By using additional data such as defined flight routes, air traffic sectors, and safety regulations, flight traffic that violates those can be identified. A goal is to find reasons or circumstances of such violations and, if possible, find and compare consequences of administrate or regulatory actions regarding to air safety violations.

Comparison of alternative scenarios: Regarding the planning of air traffic, a comparison of alternative scenarios is desired, e.g., the result of changing flight routes, or the consequences of how a new air port is designed can be examined by simulating and comparing different scenarios.

The VA framework provides specialized visualization methods which are designed to help expert users analyze the acquired movement data w.r.t. the tasks and questions mentioned above. The evaluation and interpretation of the results, however, is up to the experts and out of the scope of the software. Due to the data being retrospective in nature, visualization methods have to be designed to work on large amounts of data in order to find patterns and phenomena in past events. For a single international airport, this amounts to approx. 10,000 individual trajectories per month. Extending the scope of analysis either spatially or temporally quickly results in a massive amount of data that has to be processed.

2.2 Data set and data acquisition

The data set used in this paper consists of true 3D trajectories of airplane movements. It was recorded by RADAR tracking around an airport and includes all departing and landing IFR (instrument flight rules) flights. The RADAR tracking has a temporal resolution of 4 seconds and a range of about 50-70 km. Therefore, each trajectory has a duration of 5-10 min. with approx. 75-150 sample points per trajectory. For a duration of one month, the data set consists of a total number of 12,500 trajectories and more than 1,500,000 points.

In addition to the spatial information of the data set, the flight data is attributed with additional information per trajectory and per point. Per trajectory attributes include a flight identification (ICAO callsign), the type of aircraft, and information about the flight from the airport (i.e., arrival/departure, time stamps, and runway designations). Attributes per sample point include a time stamp, the current geographical location, as well as the current height and velocity over ground. This meta-data can be used in a visualization to convey additional information supplementing the geographical positions.

3 RELATED WORK

This section summarizes visualization strategies for spatio-temporal data: most of them are either directly two-dimensional, or use the third dimension to display information other than the spatial 3D-component of the source data, e.g., time or other non-spatial attributes of the data.

3.1 Space-time cube

The space-time cube is a visualization method developed by Hägerstrand [5] that aims at creating an intuitive visualization of spatio-temporal events. It uses a three-dimensional representation of space and time: The X- and Y-axes are used for the spatial components, while time is mapped to the Z-axis (up-axis). It is especially suitable for visualization of two-dimensional data with a temporal component.

In recent years, the space-time cube has become a popular visualization method in the field of geovisualization and geovisual analytics [6, 7], and several visualization frameworks based on the space-time cube have been developed [8, 9]. Kristensson et al. have conducted a user study on the space time cube [10], identifying trade-offs regarding the use of a space-time cube for presenting spatio-temporal data to users. Their results indicate that the space-time cube can support the comprehension of complex spatio-temporal data.

To summarize: The space-time cube directly visualizes two-dimensional movement trajectories by mapping the temporal dimension to the up-axis. To visualize three-dimensional trajectories, the space-time cube is not easily applicable: since the movement data itself is three-dimensional, all three axes are required for displaying the spatial components of the input data, therefore time cannot be mapped to one of the visual axes.

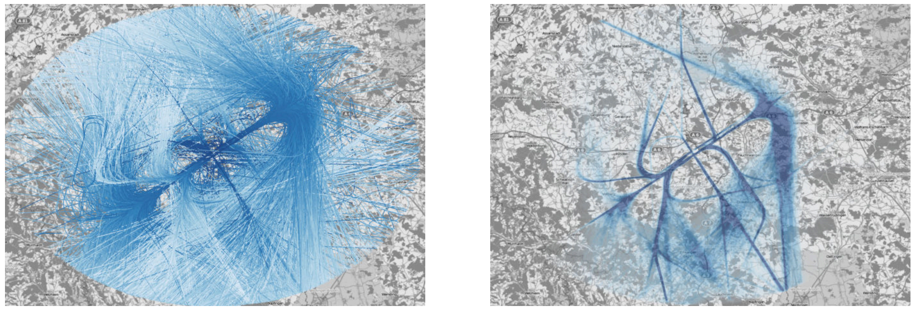
3.2 Density maps

Density maps can be used to aggregate movements using a regular grid and to represent the influence of movement trajectories on their neighborhood in the form of a spatial map. Willems et al. presented an approach for using kernel density estimation to aggregate vessel movements in a harbour [11]. They proposed the use of different kernel sizes to distinguish between historical and current movements, which enables users to identify current movements and compare them with long-time movement patterns. This way, untypical and potentially dangerous movement behavior can be detected. As an extension to this approach, Scheepens et al. introduced interactive density maps computed in realtime by utilizing modern graphics hardware. They improved their method to encode arbitrary attributes by modifying the kernel attributes and introduced an interactive approach for selecting subsets of trajectories [3].

3.3 3D density volumes

Similar to 2D density maps, the use of kernel density estimation for the aggregation of movement data can be extended to 3D. In [12], Demšar and Virrantaus have applied this method to trajectories in a space-time cube to counter the effect of visual clutter due to large numbers of trajectories being displayed. A similar approach, applied to mapping of crime scenes, can be found in [13].

While those methods use kernel density estimation to space-time-paths in a space-time cube, the same method can also directly be applied to movement trajectories that contain a spatial 3D component. This results in a density volume that contains an aggregated view to object movements in 3D - in analogy to density maps in 2D. The temporal components, however, are not preserved.



(a) Rendering of 12,500 trajectories using 2D polylines and height mapped to color.

(b) Rendering of trajectories using a 2D density map with a resolution of 128x128 px.

Fig. 1: 2D visualization of exemplary aircraft trajectories within a geovirtual environment: 2D trajectory rendering and 2D density map computed from the same data.

3.4 3D glyphs

Visualization methods can use the third dimension to display numerical values such as attributes of the input data or aggregated data. In general, numerical values are determined for a spatial position and then mapped to visual attributes of virtual objects, called glyphs, which are placed on a map [14]. This general idea of visual glyphs used on 2D maps can also be extended to 3D. Depending on the complexity of those glyphs, several visual attributes, such as color, height, and form can be used for mapping of input values. Also, complex objects can be used to visualize multiple attributes [15, 16]. In the context of movement data, glyphs are suitable for displaying static snapshots of the data, i.e., the position of objects at a specific point in time, or for displaying additional data attributes of the movement data. For visualization of large amounts of trajectories together with temporal information, however, glyphs may not be feasable, due to the large amount of objects necessary and resulting effects such as visual clutter.

3.5 Edge bundling

To reduce clutter caused by a visualization of large amounts of lines, edge bundling techniques can be used to reduce the number of individual edges visible on screen by collapsing similar edges, or parts of edges. Several edge bundling approaches exist:

Geometry-based edge clustering: Geometric edge bundling approaches as described by [17] make use of a control mesh that directs the line bundling. Such a control mesh can either be defined manually or be created automatically from the data, e.g., by using a Delaunay triangulation. Finding or computing a suitable control mesh is not always simple, but on the other hand, the benefit of this method lies in an improved control over the bundling process by exchanging or modifying the control mesh.

In [18], an edge bundling approach for the use case of geographical data is proposed. It uses a grid graph as a control mesh and bundles edges by routing them on the control grid. In the end, the result is displayed on a 3D virtual globe using smoothing curves and shading techniques to improve the visualization of bundles.

Hierarchical edge bundling: As a special case of edge bundling techniques, hierarchical edge bundling makes use of an underlying hierarchy found in the data [19]. It operates on graphs with implicit or explicit hierarchy information, i.e., compound graphs, which can be interpreted as trees. Based on this hierarchy information, curves are used to connect the nodes and create bundles. A specialized edge bundling technique suitable for circular layouts has been described by [20].

In [21], a Delaunay triangulation is used to identify the control points for creating a hierarchy using hierarchical clustering. On the resulting hierarchy a hierarchical edge bundling is performed. Inspired by progressive meshes, a progressive clustering approach has been proposed by [22].

Edge-based bundling: Another category of edge bundling algorithms works directly on graphs, without additional control meshes or hierarchy information.

Force-directed edge bundling [23] deforms the edges of a node-link diagram by applying a physical model: electrostatic forces between edges are assumed that attract edges to each other. Further, edge compatibility measures, e.g., the difference in angle or size of the two edges, are used to control the amount of bundling between edges.

Multilevel agglomerative edge bundling [24] is optimized to work on large graphs (i.e., graphs with hundred of thousands to millions of edges). It avoids the quadratical complexity (edge-to-edge) of the force-based approach by creating an edge proximity graph for each edge and performs bundling of adjacent edges based on an optimization function, which is designed to minimize the amount of ink required for plotting the node-edge diagram. Therefore, edges are joined that save the most ink when bundled. Spline rendering is used to smoothen the resulting edges.

Image-based edge bundling: In contrast to the previous edge bundling approaches, which work on the graph structure and additional data like hierarchy information or control meshes, image-based edge bundling transforms the problem into the image space.

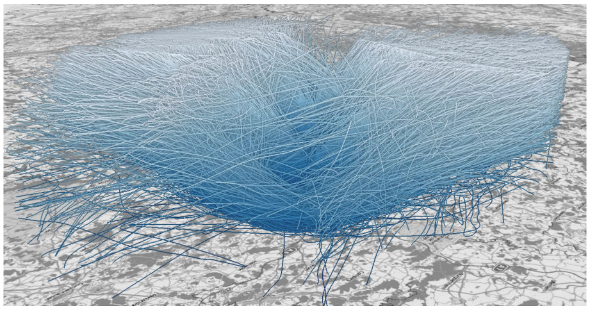
Skeleton-based edge bundling [25] uses an image-based approach to compute the centerlines of edge groups [26]. This computed skeleton is then used to move edges towards the skeleton lines and thereby bundle the edges of the graph.

As a generalized image-based approach, kernel-density-estimation-based edge bundling [27] represents edges as sample points in a texture map and uses image filtering techniques to bundle those edges in image space.

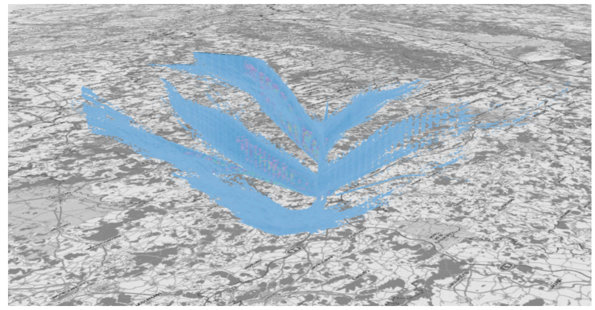
In Section 5.1, challenges and requirements for trajectory bundling as a specialized use case for edge bundling are described.

4 Visualization Approaches

This section compares and discusses properties of visualization approaches for an attributed data set of 3D trajectories in 3D geovirtual environments. The presented visualization methods are part of a visual analytics application for interactive visualization and exploration of flight movement data and are embedded into a 3D geovirtual environment based on a city and landscape model. Even though the goal is to visualize 3D movement data, not all visualization methods have to directly use all three dimensions. When exploring a data set, a 2D view to the data may be sufficient for some tasks and can provide a starting point for exploration or be used in combination with other 2D or 3D visualization methods. Therefore, we include both 2D and 3D visualization techniques as used in the context of 3D geovirtual envi-



(a) Rendering of 12,500 trajectories using 3D polylines and height mapped to color.



(b) Rendering of trajectories using a 3D density volume with a resolution of 256x256x128 px.

Fig. 2: 3D visualization of exemplary aircraft trajectories within a geovirtual environment: 3D trajectory rendering and 3D density volume computed from the same data.

ronments.

The following taxonomy is used to analyze and compare visualization methods:

Spatial information: Perception and interpretation of spatial positions are influenced by many factors, e.g., usage of perspective rendering, distortion, or the availability of a reference system. This parameter describes, how easy it is for a user to determine spatial position in a visualization.

Height perception: Visualization of heights is prone to misinterpretation due to perspective distortion, overlapping elements, or a missing reference frame. This parameter describes perception of heights in a visualization.

Occlusion and clutter: This parameter describes how much a visualization method suffers from problems such as occlusion or clutter when a large amount of data is visualized.

Cluster detection: Depending on how individual data elements are rendered, a visual detection of clusters in the resulting image may be easier or harder for a user. A visualization method can also be misleading as to creating visual clusters which do not represent real clusters in the data set. This parameter describes how accurate clusters can be detected using this visualization (presuming an appropriate color mapping).

Temporal information: This parameter describes, whether temporal information is visible at all in a visualization and how it is conveyed, i.e., directly or indirectly.

4.1 2D trajectories

Figure 1a shows a visualization of trajectories using two-dimensional lines. The height of air planes is mapped to line color (blue: 0m, white: 4000m and above). This visualization directly displays individual trajectories, taking into account only the 2D components of spatial data.

Spatial information: Spatial positions of individual trajectories in this visualization can be easily determined and interpreted due to the underlying map.

Height perception: While geographical positions of air planes can be perceived well, the color mapping and interpolation between colors make it difficult to interpret the height. Small differences between colors can mean significant distances in height, which cannot be recognized easily.

Occlusion and clutter: A large number of trajectories clutters the view, since individual lines occlude each other. This makes it difficult to identify individual trajectories or find distinctive movement patterns, e.g., outliers. Important trajectories may even be occluded completely.

Cluster detection: Since all lines are rendered on top each other, an arbitrary set of lines that is determined randomly by the rendering order is visible on top of the visualization. Therefore, the real number of lines cannot be determined. Visual detection of clusters is hardly possible and may even be misleading. Important individual trajectories (e.g., those with an untypical movement path) may also be occluded completely. Therefore, important information may not be visible to the user.

Temporal information: Temporal information of the movement is not visible with this visualization technique.

4.2 2D density maps

Figure 1b shows a 2D density map visualization of the same data. It displays an aggregated spatial view of the number of air planes crossing a 2D grid cell. The applied color mapping indicates the traffic density for each cell (white: low density, blue: high density). This visualization creates an aggregated view on the movement data but does not display individual trajectories.

Spatial information: Since density is displayed using a 2D map, the localization of density values and their spatial positions can be perceived well.

Height perception: The visualization does not contain additional height information, unless height itself is the attribute visualized using a density map.

Occlusion and clutter: Due to the aggregation of data, no visual clutter occurs but the context information of the virtual environment is occluded.

Cluster detection: In contrast to the direct rendering of trajectories, spatial clusters can be easily recognized with this visualization, because the densities of all trajectories are aggregated and visualized using a color map.

Temporal information: Temporal information is not visible, unless the density maps are recomputed for other time intervals.

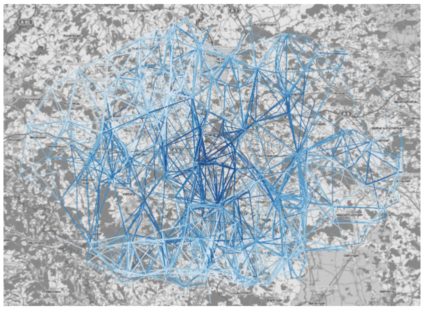
4.3 3D trajectories

In Figure 2a, the trajectories are directly visualized using 3D line rendering. Thereby, spatial 3D information of the input data is directly mapped to the visual 3D coordinate system.

Spatial information: Due to perspective distortion, positions are harder to perceive as compared to 2D trajectory visualization. Since points along a linear viewing ray are projected to the same 2D canvas position, spatial positions can hardly be distinguished. Only the combination with interactive spatial exploration, e.g., by changing position and orientation of the camera, makes it possible for a user to determine positions and distinguish seemingly overlapping trajectories from each other.

Height perception: Although heights are directly displayed in a 3D visualization, perspective distortion makes it difficult to determine the actual height of a trajectory. Relative proportions of heights, e.g., when comparing two or more trajectories, appear distorted.

Occlusion and clutter: Similar to the 2D case, a large number of trajectories lead to visual clutter. Since positions and directions of trajectories differ more due to the third dimension being used,


(a) Edge bundling of 2D trajectories with height mapped to color.

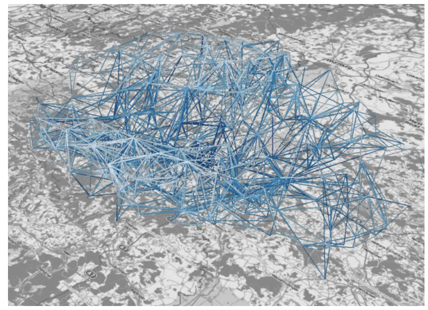

(b) Edge bundling of 3D trajectories with height mapped to color.

Fig. 3: Edge bundle of air plane trajectories in 2D and 3D using an edge-based approach.

trajectories can be distinguished better among each other, but still many trajectories on the border and top of the visualization occlude trajectories in the center.

Cluster detection: Due to the usage of all three axes, clusters can be recognized easier than in 2D. But still, occlusion and clutter may lead to misinterpretation.

Temporal information: In analogy to 2D visualization of trajectories, temporal information is not visible.

4.4 3D density volumes

Figure 2b shows the visualization of aircraft movements using 3D density volumes. Similar to 2D density maps, the influence of a trajectory to its neighborhood is computed, the resulting density values are aggregated and finally stored in a regular 3D grid. This results in a density volume that can be visualized using volume rendering techniques [28], such as direct volume rendering [29] or isosurfaces [30].

Spatial information: Similar to a direct rendering of 3D trajectories, positions are difficult to perceive due to perspective distortion and missing depth cues.

Height perception: Similar to 3D trajectories, interpretation of heights is difficult.

Occlusion and clutter: While clutter does not occur due to the aggregation of data, the problem of occlusion also exists for 3D volumes. Even small or insignificant density values at the border of the volume can occlude the density information at its center and make perception of the overall density distribution very difficult. The application of tranparency can reduce this effect, but also makes perception and interpretation of positions and their respective values even more difficult.

Cluster detection: Similar to 2D density maps, clusters can be visually recognized, but the effect of occlusion interferes.

Temporal information: Similar to 2D density maps, temporal information is not visible.

4.5 Comparison

The comparison of the chosen visualization methods with regard to the described criteria is summarized in Table 1. While spatial positions can be perceived well in 2D in general, it gets considerably harder in 3D. Heights can't be directly visualized in 2D, and are still hard to interpret even in 3D due to perspective distortion. In general, density-based approaches in both 2D and 3D do not suffer from clutter, since aggregated data instead of individual objects are visualized. However, the problem of occlusion reoccurs as soon as the method is extended to 3D. All of these visualization methods are not capable of visualizing temporal information directly. Therefore, supplementing techniques have to be used to display temporal information, such as style transfer functions (e.g., mapping time to color, size or glyphs) or interactive temporal exploration.

	Spatial	Height	Occ./Cl.	Cluster	Temp.
2D Trajectories	+	-	-	-	-
2D Density	+	-	+	+	-
3D Trajectories	o	o	-	-	-
3D Density	o	o	-	+	-

Table 1: Comparison of visualization methods regarding perception of spatial positions, perception of height, liability to occlusion and clutter, cluster detection and perception of temporal information.

5 Enhanced 3D visualization concepts

This section discusses approaches to enhance the presented visualization strategies w.r.t. the problems and challenges identified in the previous section.

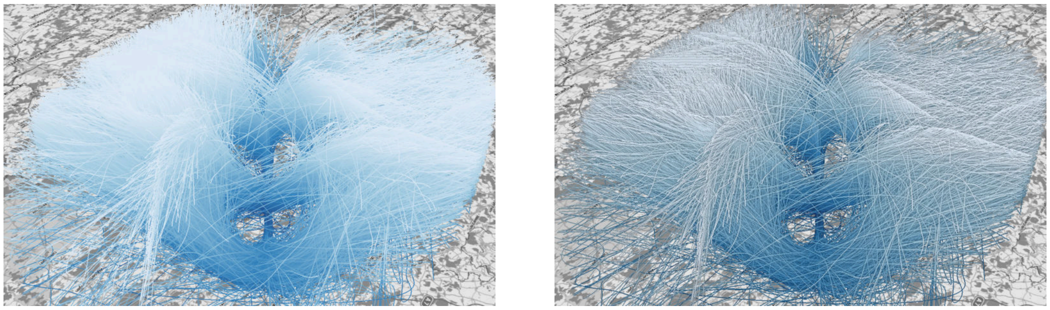
5.1 Direct trajectory visualization

For direct visualization of movement trajectories, polylines can be used to represent trajectories. However, as described before, occlusion and clutter have been identified as major problems for this kind of visualization in both the 2D and 3D cases. To compensate this, the number of simultaneously visible trajectories can be reduced. Furthermore, interactive selection and highlighting methods can be applied to improve exploration and identification of single trajectories. Also, line bundling approaches can be utilized to further reduce clutter and enhance the visual identification of clusters. On the other hand, this can distort the spatial positions of individual trajectories.

Edge bundling Many edge bundling approaches operate on graphs represented as node-link diagrams: They consist of nodes at fixed positions and edges which are modelled as straight lines between nodes. The node positions are defined either directly by a geographical position attached to the nodes or by using an initial graph layouting algorithm. Edge bundling then refines and deforms the edges, while node positions remain static.

This model is sufficient for use cases in which only the connections between nodes, but not the paths of the actual edges are important. For example, the widely used sample data set of connections between airports conveyes which airports are connected to each other, but the edges between airports do not carry any geographical meaning on their own and can therefore be transformed freely. Edge bundling in this manner has to prevent the topology of a graph, but not the position or form of edges between the nodes.

A trajectory on the other hand consists of sample points which reflect the way an object has moved. While this could formally be interpreted as a set of (unconnected) node-link diagrams, that may not be suitable for an effective bundling approach for trajectories, since the meaning of a node in this case is very different to that in a node-link diagram. If too many sample points are used, not many edges would be bundled that way, since the nodes themselves cannot be modified or merged in this approach. Too few nodes on the other hand would



(a) 3D trajectory visualization without unsharp masking.

(b) 3D trajectory visualization with unsharp masking effect for enhanced depth perception.

Fig. 4: 3D trajectory visualization with and without unsharp masking effect.

lead to a greater degree of bundling, but also distort the trajectories too much between two nodes.

In the described use case, 3D trajectories of moving air crafts need to be bundled to visualize common flight routes. For this use case, edges have to be bundled in a way that geographically near routes are merged while making sure that the path of each individual trajectory is distorted no more than to a definable threshold. Therefore, we have to assume some additional constraints and requirements for edge bundling of trajectories:

Bundling model: In contrast to edge bundling approaches that work on node-link diagrams, when bundling trajectories, nodes have to be moved during edge bundling. Since individual objects move on arbitrary paths in the investigated area, their trajectories do not all share the same points, which could be interpreted as nodes of a joined graph. This makes edge bundling harder, because such points that may become links between trajectories have to be discovered only as part of the edge bundling process.

Node positions: To preserve routes of the original trajectories, node positions must be moved as little as possible. A threshold for the movement of nodes has to be defined.

Turn angle: To maintain the characteristic of a trajectory, not only the positions but also the angles between edges have to be maintained. Therefore, the turn angle of a movement along a trajectory must be preserved.

An edge bundling algorithm for trajectories needs to satisfy these constraints. A node-link based approach would therefore have to be extended by computing a joined graph of the input trajectories that reflects the characteristic of the movement data and maintains the positions and angles of the trajectories (e.g., by using a density-based approach). Also, skeleton-based edge bundling seems very promising, but needs to be extended to 3D (see [31] for works on voxel-based 3D skeletonization).

Figure 3 shows a preliminary result of our edge bundling approach for 2D (Figure 3a) and 3D (Figure 3b) trajectories. The topology of trajectories is merged correctly, but the form of the edges is not preserved sufficiently. After bundling, the visualization can be extended by mapping the strength of a bundle (i.e., the number of trajectories merged into the bundle) to a visual attribute, such as line thickness or color.

Perception enhancements To enhance perception of positions and heights, depth cues can be provided, e.g., using lighting and shadowing techniques. Also, 2D projections of 3D lines on the ground and ledger lines can help to improve a user's interpretation of positions and heights. In addition to that, a focus+context model [32, 33] for the exploration of heights can help to identify trajectories: Trajectories can be geometrically transformed to highlight certain areas of height and thereby reveal trajectories in that area.

Figure 4 shows the effect of applying unsharp masking [34] to 3D trajectory visualization. It uses adaptive filtering to enhance the contrast of lines. This improves perception of depth in the image and helps identifying invidual lines from a large number of trajectories on the screen.

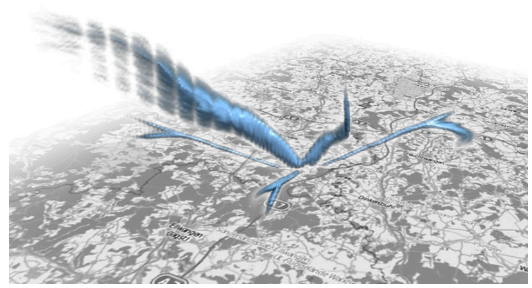


Fig. 5: Advanced 3D volume rendering with lighting and filtering using the Voreen volume renderer [35].

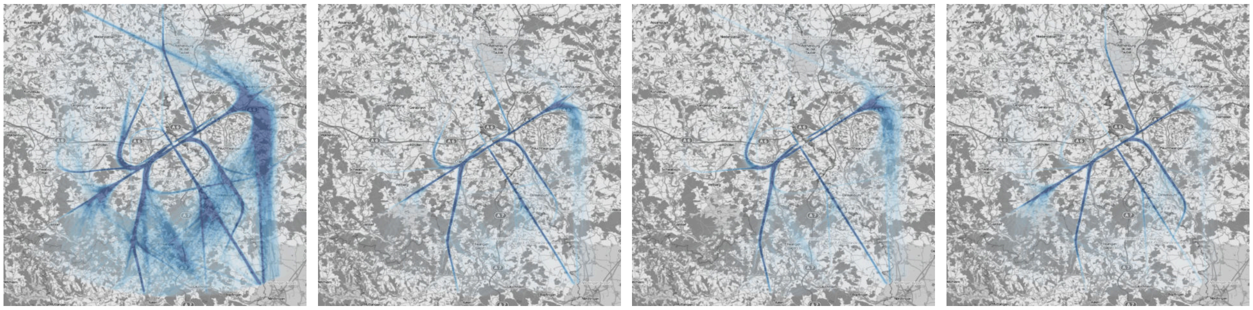
In general, more visual variables have to be used to convey data attributes of a trajectory. For example, line width and drawing style can be used as visual variables to map attributed data to. In the case of 3D trajectories, all available spatial dimensions are already used for the spatial information of the data. Therefore, future research is required for mapping the temporal dimension on visual attributes other than a spatial axis.

5.2 Density maps and volumes

For 3D density volume visualization, the main challenges are perception of positions and heights, as well as occlusion. Well known concepts for the exploration of 3D volumes, such as slicing and segmentation, can be applied. While slicing provides a spatial exploration method by defining and visualizing oriented 2D slices traversing the volume, segmentation supports detection and visualization of patterns in the density values. Similar to medical image visualization, where certain depth-ranges of an image can be selected (e.g., to visualize bones or tissues), a transfer function can be used to make volumes explorable by their depth values. Illumination and shadowing can be applied to improve perception of positions and heights (Figure 5). Also, global illumination methods such as ambient occlusion [36] can provide additional depth cues and thereby further improve perception of spatial positions.

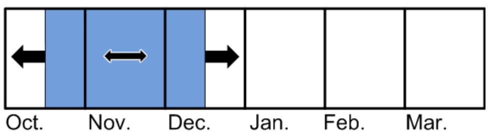
5.3 Temporal exploration

For both, direct and aggregated visualization techniques, methods are required to effectively convey and explore temporal information. As described before, this can hardly be visualized directly for 3D visualization. Therefore, indirect methods such as animation and interaction have to be applied. An interactive approach for temporal exploration of movement data seems promising, since it enables users to directly

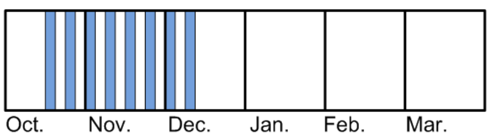


(a) Full month (october 2010). (b) 1st week (october 2010). (c) 2nd week (october 2010). (d) 3rd week (october 2010).

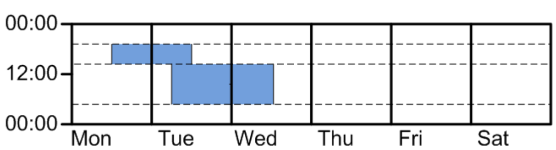
Fig. 6: Temporal exploration of trajectories from one month with a 2D density map.



(a) Time slider control for interactive selection of a single time span.



(b) Selection of repeating time intervals using graphical patterns.



(c) Flexible selection of a time span (days) and intervals inside that time span (hours per day) by using a two-dimensional layout.

Fig. 7: Time slider user interface controls.

inspect and experience temporal aspects of the movements based on the enhanced visualization techniques described before.

Interactive temporal exploration Temporal exploration can be implemented by enabling users to interactively filter input data by temporal aspects. As an example, a time slider control can be used to interactively select and modify the inspected time span. All active visualization methods, such as trajectories or density maps, will be updated using the subset of the input data that lies in the specified time span. By continously adapting and refining the time span, a user can reveal temporal aspects of the data, such as hotspots, and find temporal correlations in the inspected data.

An example is given in Figure 6: It shows the temporal exploration of a density map from trajectories of one month. In the original state, all trajectories are evaluated, producing a density map that reveals which flight paths have been used during that month. After that, the inspected time span is reduced to a week and moved dynamically over the month to discover changes in flight paths. As the visualization shows, certain routes are used more often in one week and nearly not at all in other weeks, while some flight routes are constantly used. By further refining the time span and comparing the results, it could be investigated, which flight paths are used under what circumstances.

Figure 7a shows a user interface control for the time span selection used in our framework. It consists of a timeline displayed in the background and a highlighted area in the foreground that indicates the selected time span. Start and end time can be modified by dragging either side of the selection area, thereby extending or shrinking the selected time span. By panning the area, the time span can be moved without changing its length. In analogy to spatial maps, the time line can be zoomed in and out, which changes its granularity (i.e., months, weeks, days, hours or minutes).

In addition to selecting a single time span, an examination of repeating time intervals can reveal further insight. For example, movement patterns may differ between night and day, or traffic peaks may occur always at approximately the same time of a day. Two allow this kind of examination, a user interface should allow the selection of repeating time intervals.

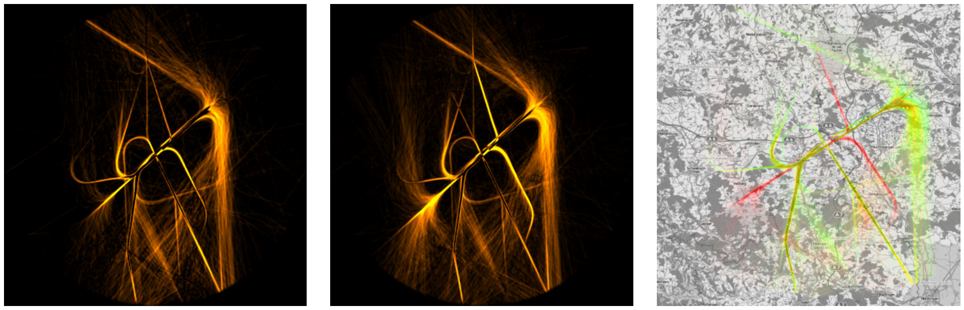
As an extension to the proposed time span control, these time intervals can be presented as graphical patterns chosen by the user. As an example, Figure 7b displays the selection of 4 days each week, repeated for an entire month. This pattern can either be defined manually by a user or be chosen from existing templates. Its repesentation as a pattern displayed inside the time line also gives a graphical indication of the used time intervals to a user.

As a more flexible approach, a two-axis visualization can be utilized: larger time intervals are selected on the X-axis (e.g., months), while the Y-axis displays smaller time intervals (e.g., days per month, or hours per day). This way, a unique pattern for the temporal selection can be defined by the user. An example of this method is shown in Figure 7c.

In such a use case, a comparison of the resulting density maps is desired. A visual comparison of density maps can be produced by computing a difference map: for each pixel it contains the difference between the density values of the original density maps. It can be visualized with the same methods as density maps, i.e., by applying an appropriate color mapping that conveys the amount of difference to a user. This method simplifies the comparison of two or more different time spans to each other. It can also be used to for example simulate different planning scenarios and compare their results.

In Figure 8, the visual comparison of density maps is shown by applying it to the density maps of individual weeks from the previous temporal exploration example. Figure 8a displays the comparison of the density maps from the first two weeks. A linear color interval is used to communicate the absolute difference value. In Figure 8b, all three original density maps are compared. Figure 8c shows an examplary visualization of difference maps embedded in a 3D geovirtual environment.

Temporal focus+context The described exploration methods can be used as a basis for a focus+context model for temporal data [37]. In general, a focus+context model distinguishes between a *focus*, i.e., a point or region-of-interest, and its *context*. In that sense, the currently selected time span defines the temporal focus for all used visualization methods. As described before, this may consist of a single time span between a given start and end point in time, but it may also be



(a) Comparison of 1st and 2nd week (october 2010). A linear color mapping is used to display the amount of change.

(b) Comparison of 1st, 2nd and 3rd week (october 2010). A linear color mapping is used to display the amount of change.

(c) Sample visualization of differences for comparison of density maps within a geovirtual environment. A multi-color mapping is used.

Fig. 8: Difference visualization of 2D density maps.

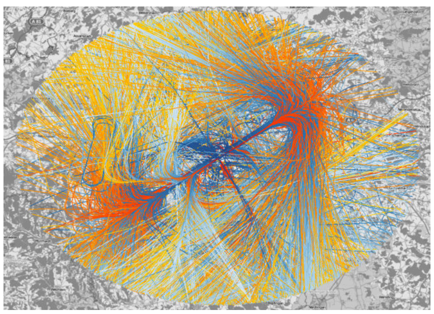


Fig. 9: Trajectory visualization with two temporal foci for comparison: 8am - 8pm (blue), 8pm - 8am (red).

defined using periodic patterns inside a time span. Data selected by this temporal focus is used as input data for the visualization methods.

The remaining data represent the temporal context. This has to be visualized in a different way, so that it is easily distinguishable from the focus. As a metaphor, the focus should be regarded as the foreground in a visualization and catch the attention of a user, while the context acts as a background. While data in the focus should be easily readable and visible in greater detail, the context can be visualized in less detail.

For direct trajectory visualization, transparency can be used to accomplish this: selected trajectories are displayed fully opaque, while trajectories in the context are semi-transparent. Also, different color schemes can be used to distinguish between focus and context. For density maps and volumes, varying kernel sizes can be used for focus and context. In addition to that, the definition of a *temporal neighborhood* can be used as a blend function between focus and context, e.g., trajectories that are near to the selected time span are more opaque than those far away.

In order to visually compare different time spans, the selection and visualization of multiple temporal foci at once can be interesting. For example, we select flights between 8am and 8pm as the first focus, flights between 8pm and 8a, as the second focus. In this visualization, time is mapped to line color, and two different color schemes are used to distinguish between the two temporal foci. The result of this comparison is shown in Figure 9.

6 Conclusions and Future Work

In this paper, spatio-temporal exploration and analysis of 3D movement trajectories of air planes have been presented as a use case. It demonstrates the need for visualization methods of true 3D movement data that require to preserve all three spatial dimensions of the input data. After exploring spatio-temporal visualization methods, it seems that many methods are specialized on 2D movements and use the third dimension for either non-spatial attributes or to map the temporal dimension. Therefore, existing methods, such as the space-time cube, are not easily applicable to such a use case.

Two different visualization approaches have been applied and examined for 2D and 3D use cases: direct visualization of trajectories and visualization of aggregated density-maps. Challenges for the visualization of movement data within 3D geovirtual environments have been identified and approaches to improve these issues were described. Most important, perception of position and height is difficult in a 3D visualization. Methods to facilitate the perception of depth, such as local and global illumination and shadowing, can be useful. In addition to that, supporting visual components such as ledger-line can be used to improve perception of spatial positions. Clutter is another problem that needs to be addressed. Filtering, selection, and higlighting can help to reduce the number of trajectories visible in a visualization and provide explorational tools to a user.

Visualization of temporal aspects in 3D visualization methods has been identified as another major challenge. When temporal information cannot be displayed directly due to missing visual dimensions, new methods to convey temporal information are required. Temporal exploration and focus+context methods can be developed to enable an interactive analysis of temporal aspects.

Acknowledgments

We wish to thank Deutsche Flugsicherung GmbH for providing the data used in this research. This work was funded by the Federal Ministry for Education and Research (BMBF), Germany within the Potsdam Research Cluster for Georisk Analysis, Environmental Change, and Sustainability (PROGRESS) and the InnoProfile Transfer research group "4DnD-Vis".

References

1. Andrienko N, Andrienko G. Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach. Springer; 2005.
2. Andrienko N, Andrienko G, Gatalsky P. Exploratory spatio-temporal visualization: an analytical review. Journal of Visual Languages and Computing. 2003 Dec;14(6):503–541.
3. Scheepens R, Willems N, van de Wetering H, et al. Interactive Density Maps for Moving Objects. IEEE Computer Graphics and Applications. 2012;32(1):56–66.
4. Elmqvist N, Tsigas P. A Taxonomy of 3D Occlusion Management for Visualization. IEEE Transactions on Visualization and Computer Graphics. 2008;14(5):1095–1109.
5. Hägerstrand TH. What about people in regional science? Papers of the Regional Science Association. 1970;p. 7–21.
6. Andrienko G, Andrienko N, Demsar U, et al. Space, time and visual analytics. International Journal of Geographical Information Science. 2010;24(10):1577–1600.

7. Kraak M. The space-time cube revisited from a geovisualization perspective. Proc 21st International Cartographic Conference. 2003;p. 10–16.
8. Andrienko N, Andrienko G, Gatalsky P. Visual data exploration using space-time cube. 21st International Cartographic Conference, Durban, 10-16 August 2003. 2003;p. 1981–1983.
9. Kraak M, Koussoulakou A. A Visualization Environment for the Space-Time-Cube. In: Fisher PF, editor. Developments in Spatial Data Handling: 11th International Symposium on Spatial Data Handling. Springer Verlag Berlin Heidelberg; 2005. p. 189–200.
10. Kristensson PO, Dahlbäck N, Anundi D, et al. An evaluation of space time cube representation of spatiotemporal patterns. IEEE Transactions on Visualization and Computer Graphics. 2009;15(4):696–702.
11. Willems N, van de Wetering H, van Wijk J. Visualization of vessel movements. Computer Graphics Forum. 2009 Jun;28(3):959–966.
12. Demšar U, Virrantaus K. Space-time density of trajectories: exploring spatio-temporal patterns in movement data. International Journal of Geographical Information Science. 2010 Oct;24(10):1527–1542.
13. Nakaya T, Yano K. Visualising Crime Clusters in a Space-time Cube: An Exploratory Data-analysis Approach Using Space-time Kernel Density Estimation and Scan Statistics. Transactions in GIS. 2010 Jun;14(3):223–239.
14. MacEachren AM. How Maps Work: Representation, Visualization and Design. Guilford Press; 1995.
15. Tominski C, Schulze-Wollgast P, Schumann H. 3D Information Visualization for Time Dependent Data on Maps. Ninth International Conference on Information Visualisation (IV'05);p. 175–181.
16. Sidharth T, Hanson AJ. A 3D Visualization of Multiple Time Series on Maps. 14th International Conference Information Visualisation. 2010;p. 336–343.
17. Cui W, Zhou H, Qu H, et al. Geometry-Based Edge Clustering for Graph Visualization. IEEE Transactions on Visualization and Computer Graphics. 2008 Nov;14(6):1277–1284.
18. Lambert A, Bourqui R, Auber D. 3D Edge Bundling for Geographical Data Visualization. In: Proceedings of the 2010 14th International Conference Information Visualisation. IV '10. Washington, DC, USA: IEEE Computer Society; 2010. p. 329–335.
19. Holten D. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data. IEEE Transactions on Visualization and Computer Graphics. 2006 Sep;12(5):741–748.
20. Gansner ER, Koren Y. Improved circular layouts. In: Proceedings of the 14th international conference on Graph drawing. GD'06. Berlin, Heidelberg: Springer-Verlag; 2007. p. 386–398.
21. Zhou H. Energy-Based Hierarchical Edge Clustering of Graphs; 2008. .
22. Qu H, Zhou H, Wu Y. Controllable and progressive edge clustering for large networks. In: Proceedings of the 14th international conference on Graph drawing. GD'06. Berlin, Heidelberg: Springer-Verlag; 2007. p. 399–404.
23. Holten D, Van Wijk JJ. Force-Directed Edge Bundling for Graph Visualization. Computer Graphics Forum. 2009;28(3):983–990.
24. Gansner ER, Hu Y, North S, et al. Multilevel agglomerative edge bundling for visualizing large graphs. In: Proceedings of the 2011 IEEE Pacific Visualization Symposium. PACIFICVIS '11. Washington, DC, USA: IEEE Computer Society; 2011. p. 187–194.
25. Ersoy O, Hurter C, Paulovich F, et al. Skeleton-based edge bundling for graph visualization. IEEE TVCG 17(12) (Proc InfoVis). 2011;.
26. Telea A, Ersoy O. Image-Based Edge Bundles: Simplified Visualization of Large Graphs. Computer Graphics Forum. 2010;29(3):843–852.
27. Hurter C, Ersoy O, Telea A. Graph Bundling by Kernel Density Estimation. Computer Graphics Forum. 2012 Jun;31(3pt1):865–874.
28. Drebin RA, Carpenter L, Hanrahan P. Volume rendering. SIGGRAPH Comput Graph. 1988 Jun;22(4):65–74.
29. Callahan SP, Callahan JH, Scheidegger CE, et al. Direct Volume Rendering: A 3D Plotting Technique for Scientific Data. Computing in Science and Engineering. 2008;10(1):88–92.
30. Hadwiger M, Sigg C, Scharsach H, et al. Real-Time Ray-Casting and Advanced Shading of Discrete Isosurfaces. Computer Graphics Forum. 2005;24(3):303–312.
31. Reniers D, van Wijk J, Telea A. Computing Multiscale Curve and Surface Skeletons of Genus 0 Shapes Using a Global Importance Measure. IEEE Transactions on Visualization and Computer Graphics. 2008;14:355–368.
32. Furnas GW. Generalized Fisheye Views. Proceedings of CHI. 1986;p. 16–23.
33. Leung YK, Apperley MD. A Review and Taxonomy of Distortion-Oriented Presentation Techniques. ACM Transactions on Computer-Human Interaction. 1994;1(2):126–160.
34. Polesel A, Ramponi G, Mathews VJ. Image Enhancement via Adaptive Unsharp Masking. IEEE Trans Image Processing. 2000;9:505–510.
35. Ropinski T, Meyer-Spradow J, Diepenbrock S, et al. Interactive Volume Rendering with Dynamic Ambient Occlusion and Color Bleeding. Computer Graphics Forum. 2008;27(2):567–576.
36. Pharr M, Green S. Ambient occlusion. In: Fernando R, editor. GPU Gems. Addison-Wesley; 2004. p. 667–692.
37. Carvalho A, de Sousa AA, Ribeiro C, et al. A temporal focus + context visualization model for handling valid-time spatial information. Information Visualization. 2008 Jun;7(3):265–274.